

# **Manual de Desarrollador Externo**

**Envío Automático de Archivos Electrónicos de Cesión  
(AEC)**

**Oficina Factura Electrónica  
Subdirección Informática  
Servicio Impuestos Internos**

**Fecha:31/01/2007**

## INDICE

<b>INTRODUCCIÓN .....</b>	<b>3</b>
<b>CAPITULO 1.....</b>	<b>4</b>
OBJETIVOS Y CARACTERISTICAS .....	4
1.1 OBJETIVOS DE LA APLICACIÓN.....	4
1.2 CARACTERÍSTICAS DEL PROGRAMA CLIENTE .....	4
1.3 AUTENTICACIÓN Y PERMISOS DE UPLOAD.....	4
<b>CAPITULO 2.....</b>	<b>5</b>
DEFINICIÓN DEL REQUEST.....	5
2.1 PARÁMETROS NECESARIOS PARA EL ENVÍO.....	5
2.2 EJEMPLO DE MENSAJE CLIENTE (REQUEST) .....	5
<b>CAPITULO 3.....</b>	<b>7</b>
DESCRIPCIÓN DEL RESPONSE.....	7
3.1 PARÁMETROS DE SALIDA .....	7
3.2 VALORES POSIBLES DEL PARÁMETRO STATUS .....	7
3.3 EJEMPLOS DE RESPONSE.....	7
<b>CAPITULO 4.....</b>	<b>9</b>
DIRECCIONES PARA EL ENVÍO AUTOMÁTICO.....	9
4.1 CERTIFICACIÓN .....	9
4.2 PRODUCCIÓN.....	9
<b>ANEXO 1.....</b>	<b>10</b>
PROBLEMAS FRECUENTES DETECTADOS EN LA IMPLEMENTACIÓN DEL ENVÍO.....	10

## INTRODUCCIÓN

Este documento está dirigido a quienes tengan la misión de implementar el envío automático de Archivos Electrónicos de Cesión (AEC), en adelante “Programa Cliente”.

Para la implementación del envío automático, el programa cliente deberá ser capaz de realizar las operaciones de Autenticación mediante Certificado Digital y de UPLOAD de archivos, las que normalmente son realizadas a través del browser.

Este manual aborda el proceso de UPLOAD. Respecto a la Autenticación, para poder realizar cualquier operación con el SII, los clientes deben estar previamente autenticados a través de la [AUTENTICACIÓN AUTOMÁTICA](#).

### **Requisitos de uso.**

Para poder utilizar este manual, es necesario tener previo conocimiento de Formato [\(RFC 1867\)](#), XML, Web Services y Certificado Digital.

## **CAPITULO 1**

### **OBJETIVOS Y CARACTERISTICAS**

#### **1.1 Objetivos de la aplicación**

Esta aplicación forma parte del sistema Registro Electrónico de Cesión de Créditos (RPETC), y permite enviar Archivos Electrónicos de Cesión (AEC), mediante UPLOAD del archivo.

El objetivo del programa cliente es realizar el envío automático de AEC, para esto se deberá construir un programa que simule un browser, es decir debe realizar la comunicación tipo Socket de TCP/IP con encriptación de datos SSL

#### **1.2 Características del programa Cliente**

Algunas de las características que debe el programa cliente son:

- ✓ Debe ser una aplicación P2P.
- ✓ Debe establecer conexión vía Socket con la aplicación definida por el SII para el UPLOAD de AEC.
- ✓ Debe ser capaz de realizar autenticación automática y UPLOAD de archivos.
- ✓ Debe ser capaz de recibir la respuesta del SII (Response) y procesarla.

#### **1.3 Autenticación y Permisos de Upload**

- ✓ Solo las empresas autorizadas DTE pueden realizar envíos automáticos de AEC.
- ✓ Para realizar un envío de AEC contribuyente autenticado, debe tener representación sobre la empresa cedente, y tener permiso de "Envío de Documentos".

## CAPITULO 2

### Definición del Request

El request que debe formar la aplicación cliente se detalla a continuación:

#### 2.1 Parámetros necesarios para el Envío

El siguiente cuadro muestra los parámetros requeridos en el Request:

Parámetro	Descripción
Token	Cookie de autenticación. Debe ir en el header del request.
emailNotif	Mail al que se debe notificar el resultado del envío.
rutCompany	Rut empresa Cedente
dvCompany	Digito Verificador Empresa Cedente
archivo	Archivo AEC.

#### 2.2 Ejemplo de mensaje Cliente (Request)

**Nota:** Lo destacado con rojo corresponde al Header.

```
POST /cgi_rtc/RTC/RTCAnotEnvio.cgi HTTP/1.0^M
Accept: image/gif, image/x-xbitmap, image/jpeg, image/pjpeg,
application/vnd.ms-powerpoint, application/ms-excel, application/msword,
*/**^M
Referer: {URL que referencia a upload Ej. http://empresaabc.cl/test.html}^M
Accept-Language: es-cl^M
Content-Type: multipart/form-data: boundary={boundary data: Ej. -----
-----7d23e2a11301c4}^M
Accept-Encoding: gzip, deflate^M
User-Agent: Mozilla/4.0 (compatible; PROG 1.0; Windows NT 5.0; YComp
5.0.2.4)^M
Host: {Host Id. Ej: https://maullin.sii.cl}^M
Content-Length: {largo total de mensaje sin Req. Header. Ej.: 10240}^M
Connection: Keep-Alive^M
Cache-Control: no-cache^M
Cookie: TOKEN={Entregado por Autenticación. Ej.: YZD0II2ApZjLM}^M
^M
{Comienzo de Multipart/Form-data}
-----7d23e2a11301c4^M
Content-Disposition: form-data; name="emailNotif"^M
^M
correo@empresa.cl^M
-----7d23e2a11301c4^M
Content-Disposition: form-data; name="rutCompany"^M
^M
88888888^M
-----7d23e2a11301c4^M
Content-Disposition: form-data; name="dvCompany"^M
^M
8^M
-----7d23e2a11301c4^M
Content-Disposition: form-data; name="archivo"; filename="EnvioEjemploAEC.xml" ^M
```

```
Content-Type: text/xml^M
^M
<?xml version="1.0" encoding="ISO-8859-1"?>^M
<AEC xmlns="http://www.sii.cl/SiiDte" xmlns:ds="http://www.w3.org/2000/09/xmldsig#"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" version="1.0" xsi:schemaLocation="http://www.sii.cl/SiiDte
AEC_v10.xsd">^M
  <DocumentoAEC ID="DOCAEC1">^M
    <Caratula version="1.0">^M
      . . . . .
      . . . . .
      . . . . .
    </Signature>^M
  </AEC>^M
^M
-----7d23e2a11301c4--^M
```

## CAPITULO 3

### Descripción del Response

#### 3.1 Parámetros de Salida

Una vez que el SII ha recepcionado el archivo AEC, nuestro sistema genera una respuesta en formato XML, que indica entre otras cosas el estado de la recepción del archivo.

El siguiente cuadro muestra los parámetros entregados por el SII:

Parámetro	Descripción
<RUTSENDER>	Rut y Dv de usuario autenticado.
<RUTCOMPANY>	Rut y Dv de la Empresa Cedente.
<FILE>	Nombre del archivo recibido, no incluye path.
<TIMESTAMP>	Fecha y Hora de recepción del AEC.
<STATUS>	Resultado de la operación.
<TRACKID>	Número de recepción asignado al envío. Este se entrega solo si el resultado del envío resultó OK (status = 0).

#### 3.2 Valores posibles del parámetro STATUS

Estado	Descripción
0	Envío recibido OK.
1	Rut usuario autenticado no tiene permiso para enviar en empresa Cedente.
2	Error en tamaño del archivo enviado.
4	Faltan parámetros de entrada.
5	Error de autenticación, TOKEN inválido, no existe o está expirado.
6	Empresa no es DTE.
9	Error Interno.
10	Error Interno

#### 3.3 Ejemplos de Response

##### 3.3.1 Response OK

```
<?xml version="1.0"?>
<RECEPCIONAEC>
  <RUTSENDER>1-9</RUTSENDER>
  <RUTCOMPANY>88888888-8</RUTCOMPANY>
  <FILE>AECautomOK.xml</FILE>
  <TIMESTAMP>29-11-2006 14:50:13</TIMESTAMP>
  <STATUS>0</STATUS>
  <TRACKID>0000038936</TRACKID>
</RECEPCIONAEC>
```

### 3.3.2 Response con Error de Autenticación

```
<?xml version="1.0"?>
<RECEPCIONAEC>
  <RUTSENDER>-</RUTSENDER>
  <RUTCOMPANY>88888888-8</RUTCOMPANY>
  <FILE>AECautomOK.xml</FILE>
  <TIMESTAMP>29-11-2006 15:49:08</TIMESTAMP>
  <STATUS>5</STATUS>

</RECEPCIONAEC>
```

### 3.3.3 Response con Error de Permisos.

```
<?xml version="1.0"?>
<RECEPCIONAEC>
  <RUTSENDER>1-9</RUTSENDER>
  <RUTCOMPANY>77777777-7</RUTCOMPANY>
  <FILE>AECautomOK.xml</FILE>
  <TIMESTAMP>29-11-2006 15:52:04</TIMESTAMP>
  <STATUS>1</STATUS>

</RECEPCIONAEC>
```



## **CAPITULO 4**

### **Direcciones para el Envío Automático**

#### **4.1 Certificación**

Para realizar pruebas de UPLOAD automático, el programa cliente que simula un Browser, debe conectarse al sitio de certificación del SII. Los siguientes son los datos para realizar la conexión:

- ✓ Host : `maullin.sii.cl`
- ✓ Puerto : 443
- ✓ URL : `/cgi_rtc/RTC/RTCAnotEnvio.cgi`

#### **4.2 Producción**

Para envíos automáticos a producción, los datos de conexión son:

- ✓ Host : `palena.sii.cl`
- ✓ Puerto : 443
- ✓ URL : `/cgi_rtc/RTC/RTCAnotEnvio.cgi`

## ANEXO 1

### Problemas Frecuentes Detectados en la Implementación del Envío.

Algunos de los problemas frecuentes detectados en el envío se relacionan con:

Campo	Problema	Solución
boundary	Diferencia en la composición del <b>boundary</b> (Header y Cuerpo del Mensaje)	<p>El boundary del “<b>Cuerpo del Mensaje</b>”, debe llevar <b>dos caracteres “más”</b> que los caracteres del boundary <b>del Header</b>. Además, el boundary, del <b>Final de mensaje</b>, debe llevar dos guiones al comienzo y dos guiones al final, antes del control M (^M). Ver Ejemplos.</p> <p>1-Ejemplo boundary: <b>Header</b></p> <pre>Content-Type: multipart/form-data; boundary=7d23e2a11301c4^M</pre> <p>2-Ejemplo boundary: <b>Cuerpo del mensaje</b>, debe tener 2 guiones al inicio.</p> <pre>--7d23e2a11301c4^M</pre> <p>3-Ejemplo boundary: <b>Final de mensaje</b>, debe llevar dos guiones del comienzo más dos adicionales antes del control M (^M).</p> <pre>--7d23e2a11301c4--^M</pre>
Largo del Mensaje	El largo del mensaje incluye datos del Header	<p>El largo del mensaje no debe incluir datos del Header, tal como se muestra en la figura 2.2, el Header corresponde a los datos encerrado en el recuadro, la línea “<b>{Comienzo de Multipart/Form-data}</b>”, corresponde a una línea en blanco, por lo que el inicio del mensaje comenzaría en el primer carácter o guión después de “<b>{Comienzo de Multipart/Form-data}</b>”</p> <p>Ejemplo:</p> <pre>{Comienzo de Multipart/Form-data} -7d23e2a11301c4^M</pre> <p>Donde el circulo indica el inicio del mensaje.</p>
Host	Uso del campo Host en programas C y que además están utilizando la librería de OpenSSL	Los programas en lenguaje C, los cuales además utilizan la librería OpenSSL, “ <b>NO deben incluir el Campo Host</b> ”.
Control M (^M)	Falta el Control M	<p>- Los caracteres Control M (^M), forman parte del archivo, por lo que estos deben ser enviados.</p> <p>- La forma de parsear los Control M (^M), depende del de Cliente:</p> <ul style="list-style-type: none"><li>• Si el Cliente es un programa en C++ y utiliza el <b>objeto string</b> para la concatenación, se debe indicar cada fin de línea con “\r\n”, para parsear el ^M.</li><li>• Si el Cliente es un programa en C o C++ y utiliza <b>char</b>, se debe indicar cada fin de línea con “\n”.</li></ul>